

Generative AI for Programming Education: Benchmarking ChatGPT, GPT-4, and Human Tutors

Year: 2023 | Citations: 115 | Authors: Tung Phung, Victor-Alexandru Pădurean, J. Cambroner, Sumit Gulwani, Tobias Kohn

Abstract

Generative AI and large language models hold great promise in enhancing computing education by powering next-generation educational technologies. State-of-the-art models like OpenAI's ChatGPT [8] and GPT-4 [9] could enhance programming education in various roles, e.g., by acting as a personalized digital tutor for a student, a digital assistant for an educator, and a digital peer for collaborative learning [1, 2, 7]. In our work, we seek to comprehensively evaluate and benchmark state-of-the-art large language models for various scenarios in programming education. Recent works have evaluated several large language models in the context of programming education [4, 6, 10, 11, 12]. However, these works are limited for several reasons: they have typically focused on evaluating a specific model for a specific education scenario (e.g., generating explanations), or have considered models that are already outdated (e.g., OpenAI's Codex [3] is no longer publicly available since March 2023). Consequently, there is a lack of systematic study that benchmarks state-of-the-art models for a comprehensive set of programming education scenarios. In our work, we systematically evaluate two models, ChatGPT (based on GPT-3.5) and GPT-4, and compare their performance with human tutors for a variety of scenarios in programming education. These scenarios are designed to capture distinct roles these models could play, namely digital tutors, assistants, and peers, as discussed above. More concretely, we consider the following six scenarios: (1) program repair, i.e., fixing a student's buggy program; (2) hint generation, i.e., providing a natural language hint to the student to help resolve current issues; (3) grading feedback, i.e., grading a student's program w.r.t. a given rubric; (4) peer programming, i.e., completing a partially written program or generating a sketch for the solution program; (5) task creation, i.e., generating new tasks that exercise specific types of concepts or bugs; (6) contextualized explanation, i.e., explaining specific concepts or functions in the context of a given program. Our study uses a mix of quantitative and qualitative evaluation to compare the performance of these models with the performance of human tutors. We conduct our evaluation based on 5 introductory Python programming problems with a diverse set of input/output specifications. For each of these problems, we consider 5 buggy programs based on publicly accessible submissions from [geeksforgeeks.org](https://www.geeksforgeeks.org) [5] (see Figure 1); these buggy programs are picked to capture different types of bugs for each problem. We will provide a detailed analysis of the data and results in a longer version of this poster. Our preliminary results show that GPT-4 drastically outperforms ChatGPT (based on GPT-3.5) and comes close to human tutors' performance for several scenarios.